Sphinx: Monte Carlo Program for Polarised Nucleon-Nucleon Collisions

Stefan Güllenstern
Max-Planck-Institut für Kernphysik, Postfach 103 980
D-69029 Heidelberg, Germany

Paweł Górnicki
Institute of Physics, Polish Academy of Sciences
and
Center for Theoretical Physics, Polish Academy of Sciences
Al. Lotników 32/46, PL-02-668 Warsaw, Poland

Lech Mankiewicz
Nicolaus Copernicus Astronomical Center, ul. Bartycka 18
PL-00-716 Warsaw, Poland

Andreas Schäfer
Institut für Theoretische Physik, Universität Frankfurt
D-60054 Frankfurt am Main, Germany

February 7, 2008

PROGRAM SUMMARY

Title of program: Sphinx – HEP MONTE CARLO

Catalogue number:

Program available from:

Computer: DEC Station 5000/260; other machines with `FORTRAN 77` and sufficient capacity should be able to run this program

Installation: Institut for Theoretical Physics, University of Frankfurt, Frankfurt/Main

Operating system: tested under `UNIX` but does not depend on the particular operating system

Programing language used: `FORTRAN 77`

High speed storage required: 5 Mbytes

Number of bits per byte: 8

Number of lines in combined test deck: 19296

CPC subprograms used: JETSET 7.4

Keywords: polarized nucleon-nucleon scattering, high energy physics, Monte Carlo simulation

Nature of physical problem: This program can be used to simulate polarised nucleon - nucleon collisions at high energies. Spins of colliding particles are taken into account. The program allows to calculate cross sections for various processes.

Method of solution: The existing Monte Carlo program PYTHIA 5.7 has been modified to incorporate spin effects. The program incorporates all features of PYTHIA.

Restrictions on the complexity of the problem: Spins of colliding nucleons must be parallel to the collision axis.

Typical running time: $\approx$ 0.1 sec CPU per event on DEC Station 5000/260

References: Reader has to refer to PYTHIA manual [1] for additional details. Further informations can be obtained from: Prof. A. Schäfer, Institut für Theoretische Physik, Universität Frankfurt a. M., Robert Mayer-Strasse 10, D-60054 Frankfurt am Main, Germany

LONG WRITE-UP

1 Introduction

The detailed investigation of hadronic spin effects, most notably in deep-inelastic lepton-nucleon scattering, has proven to allow for extremly sensitive tests of QCD. One of the most interesting issues is the possible contribution of the axial-vector anomaly. This anomaly plays a central role in modern quantum-field-theory and spin effects might offer the only possibility to actually detect it. To estimate its contribution one has to know the distribution function for polarised gluons $\Delta g(x, Q^2)$. It is argued that the total spin carried by gluons could be large, because $\alpha_s(Q^2)\Delta g(Q^2)$, with $\Delta g(Q^2) = \int_0^1 \mathrm{d}x\ \Delta g(x, Q^2)$, is a renormalisation group invariant, such that if $\Delta g(Q^2)$ is a few percent at the constituent quark level it could be very sizeable at large Q^2. In deep-inelastic scattering it is impossible to distinguish between the regular quark contribution and the gluon contribution via the anomaly. There is actually no conceptually sound way to separate them. In exclusive or semi-inclusive lepton-hadron scattering channels it is in principle possible to discern a gluonic and a quark contribution, but the by far most direct and conceptually clean way are polarised proton-proton collisions. RHIC [2] offers the option to study polarised proton collisions up to $\sqrt{s} = 300$ GeV which would be ideal for this purpose. To analyse this option in detail, to plan the experiments, and to determine the reachable precission a detailed Monte Carlo program is needed, which we provide with this publication. This program can handel, however, only longitudinal polarisation. All polarisation effects in proton-proton reactions share the same problem. As only a small fraction of the partons is polarised the signal to background ratios are typically of the order percent, such that a detailed simulation is necessary to extract quantitative results.

Because a state-of-art Monte Carlo program is usually the result of many years of development we modified an existing, widly used program called Pyhtia [1]. We have been interested only in those parts of the program that correspond to nucleon-nucleon collisions. Consequently only these routines were changed. The resulting polarised version of Pythia is named Sphinx, an acronym for **S***imulator of* **P***olarised* **H***adronic* **IN***teract(***X***)ions.*

It is interesting to note that Sphinx is able to function in unpolarised

mode as well. In that case it is functionally equivalent to Pythia. This means that results will be the same but because some data structures inside the program have been modified the program requires more space.

The main problem was to introduce the changes in a way compatible with the basic structure of Pythia, e. g. the way in which Pythia evaluates the cross sections or simulates the effects of higher order QCD corrections. Pythia is based on the parton model. The main assumption is that the nucleon may be represented by an incoherent mixture of quarks and gluons carrying portions of nucleon momentum. Their momentum distributions are described by parton distributions. In the parton model the nucleon is composed of quarks and antiquarks of various flavours and of gluons. In the polarised case each parton comes in one of two helicities: righthanded and lefthanded. Therefore the number of parton components has to be doubled, corresponding to the two degrees of freedom for longitudinal polarisation. The most natural idea was to generalize the notion of flavour to account for the polarisation. So instead of d-quark we have lefthanded d-quark and righthanded d-quark etc. Because in Pythia gluons are treated as a special "flavour" the scheme applies to them as well. As far as the general structure of the program is concerned the only complication of the polarised code is that we have more "flavours" and this is rather easy to implement.

Polarisation is followed in Sphinx only up to the hard partonic interaction, i. e. the hadronic cross section and the higher order corrections in the initial state shower are evaluated spin*dependently* whereas all final state interactions as fragmentation and decays are treated spin*independently*. No theoretical model for polarised fragmentation exists. It is, however, generally accepted that it should not depend on longitudinal polarisation. For transverse polarisation the situation would, however, be different. One knows experimentally that substantial effects exist, e. g. a strong coupling between transverse spin and transverse momentum.

The polarisation effects have been implemented in the following parts of the program:

1. Parton distributions. At the moment we supply six sets of polarised parton distributions: Altarelli-Stirling [3], two parametrisations from a Ross-Roberts article [4] and three sets from Gehrmann&Stirling [6]. First order Altarelli-Parisi [5] evolution is taken into account in all cases.

2. Hard processes. The processes currently implemented in the polarised mode are summarized in Table 1. Other processes may be used as well but the results will be averaged over polarisations. The cross sections for all polarised processes are given in the literature [7], and we checked them.

3. Initial state showering. The polarised case was not discussed in this context before. We obtained all necessary formulae.

4. Documentation. The event listing has been modified such that it displays polarisation information about the interacting particles up to the hard partonic interaction.

Hereafter we do not try to explain the structure of PYTHIA 5.7 and we indicate only the most significant modifications which lead to SPHINX. Readers unfamiliar with PYTHIA should first consult the appropriate manual [1].

2 From PYTHIA to SPHINX: modifications

Our program is an extension of PYTHIA version 5.7. The PYTHIA 5.7 program heavily relies on JETSET 7.4 subroutines [8]. These subroutines are mainly used to simulate processes that take place after the hard partonic interaction, i.e. fragmentation and decays, or provide spinindependent manipulations as for example Lorentz-transformations. Because polarisation effects in the final state are neglected in SPHINX, the JETSET program could be left untouched. Only the set-up of an event listing which is also a task of JETSET in a PYTHIA run has been taken over from SPHINX in the polarised mode. The event listing in SPHINX provides then in addition information about the polarisation flow. For that purpose two new subroutines have been added which are modifications of the corresponding JETSET routines (see below). By this means SPHINX provides a suitable interface to JETSET such that the programs may be linked without any modifications in the latter. There is only one exception: the JETSET subroutine `LUGIVE`, which returns the values of all JETSET *and* PYTHIA common block variables cannot be longer used, because in SPHINX the dimensions of some PYTHIA arrays have been enlarged so that the formats no longer match.

Table 1: List of processes implemented in the polarised mode.

`ISUB`	Process	Comment
1	$q_i\bar{q}_j \to \gamma^*/Z^0$	quark-antiquark annihilation into virtual γ^*/Z^0
2	$q_i\bar{q}_j \to W^\pm$	annihilation into charged vector boson
11	$q_iq_j \to q_iq_j$	(anti-)quark – (anti-)quark scattering; annihilation diagram not included
12	$q_i\bar{q}_i \to q_k\bar{q}_k$	annihilation process
13	$q_i\bar{q}_i \to gg$	annihilation into gluon pair
14	$q_i\bar{q}_i \to g\gamma$	annihilation into gluon and prompt γ
15	$q_i\bar{q}_i \to gZ^0$	annihilation into gluon and Z^0
16	$q_i\bar{q}_i \to gW^\pm$	annihilation into gluon and $W^\pm$
18	$q_i\bar{q}_i \to \gamma\gamma$	annihilation into γ pair
19	$q_i\bar{q}_i \to \gamma Z^0$	annihilation into γ and Z^0
20	$q_i\bar{q}_i \to \gamma W^\pm$	annihilation into γ and $W^\pm$
28	$q_ig \to q_ig$	(anti-)quark – gluon scattering
29	$q_ig \to q_i\gamma$	prompt γ production in (anti-)quark – gluon scattering
30	$q_ig \to q_iZ^0$	Z^0 production in (anti-)quark – gluon scattering
31	$q_ig \to q_jW^\pm$	$W^\pm$ production in (anti-)quark – gluon scattering
53	$gg \to q_k\bar{q}_k$	gluon fusion
68	$gg \to gg$	gluon – gluon scattering

In many places inside PYTHIA partonic data are stored in arrays indexed by flavour. Inclusion of helicities means doubling the number of "flavours" as described in Section 1. For that reason we had to enlarge the arrays. The unpolarised mode has been preserved throughout the program and works exactly as the original code. All modifications are implemented in a treelike structure with several branching points where the user may choose between the polarised and unpolarised modes. The default is always unpolarised. An interesting point is that one can combine polarised and unpolarised treatments for different processes and for various stages of the same process. This has to be done with care and one has to be aware of this not always being justified from the physical point of view. The branching points were implemented by means of the `IF-ELSE` structure which directs the program flow according to a specific switch. Each functional part (usually a subroutine) has its own *local* variable called `IPOL`. This variable controls the mode and its value depends on some parameters[1] `MSTP(x)` and `NSUB(x)` to be defined by the user in the main program. For example, it is possible to run the code with polarised partonic cross section and polarised parton distributions but unpolarised initial state showering etc. This solution provides flexibility but has to be used carefully. The major disadvantage of `IF-ELSE` constructions is that parts of the code are multiplied. This makes servicing of the code more cumbersome because one has to introduce the same change in different places at once. In addition the resulting code became quite lengthy.

SPHINX is as PYTHIA a "slave-system", i. e. it consists only of callable subroutines where two of them (`PYINIT` and `PYEVNT`) have to be called by the user to perform the event generation and a few other, e. g. `DPLIST`, `PYSTAT`, etc., could be called to obtain further event information. The rest of the subroutines is of internal use. Therefore the user has to supply a main program where all relevant parameters and switches have to be specified and these subroutines have to be called. Because this structure is the same as in PYTHIA the reader is referred to [1] again for details about the general PYTHIA parameters. In this article only the new parameters in SPHINX are discussed. The modifications made in the individual subroutines are described in Section 3. In Table 1 we list all new parameters introduced to control the polarised mode. Finally in Section 4 examples for a main program and the result of the corresponding test runs are presented.

[1]See Table 1

Table 2: Parameters controlling the polarised mode

Parameter		Description	Default
`MSTP(171)`		beam polarisation	0
	=0:	unpolarised	
	=1:	polarisation in $+z$ direction	
	=2:	polarisation in $-z$ direction	
`MSTP(172)`		target polarisation	0
	=0:	unpolarised	
	=1:	polarisation in $+z$ direction	
	=2:	polarisation in $-z$ direction	
`MSTP(175)`		use of polarised parton distribution in polarised initial state shower (`MSTP(176)=1`)	1
	=0:	unpolarised distribution; for testing only, *do not use!*	
	=1:	polarised distribution	
`MSTP(176)`		initial state showering mode	0
	=0:	unpolarised	
	=1:	polarised	
`MSTP(177)`		set of polarised parton distributions Δq and Δg used; in addition one has to specify unpolarised set as in standard Pythia	0
	=0:	$\Delta q = 0$ and $\Delta g = 0$ (no polarisation)	
	=1:	fake polarisation, built up from unpolarised distribution according to $\Delta q = \frac{\texttt{MSTP(178)}}{100} q$	
	=2:	Altarelli-Stirling parametrization [3]; data file `altsti.dat` required	
	=3:	Ross-Roberts parametrization [4] set d	
	=4:	Ross-Roberts parametrization [4] set a; data file `rosroa.dat` required	
	=5:	Gehrmann-Stirling parametrization [6] set a; data file `partons.dat` required	
	=6:	Gehrmann-Stirling parametrization [6] set b; data file `partons.dat` required	
	=7:	Gehrmann-Stirling parametrization [6] set c; data file `partons.dat` required	
	=8:	fake polarisation, for testing only, *do not use!*	

Table 2: Parameters controlling the polarised mode, continued

Parameter		Description	Default
MSTP(178)		percentage of fake polarisation for MSTP(177)=1	0
MSTP(180)		mode selection (master switch)	0
	=0:	unpolarised mode; this value overrides all other polarisation switches	
	=1:	polarised mode	
NSUB(ISUB)		mode for subprocess ISUB	0
	=0:	unpolarised treatment	
	=1:	polarised treatment	
NSEL		menue of polarised processes	0
	=1:	ISUB = 11,12,13,28,53,68 switched on	
	=10:	ISUB = 14,18,29 switched on	
	=11:	ISUB = 1 switched on	
	=12:	ISUB = 2 switched on	
	=13:	ISUB = 15,30 switched on	
	=14:	ISUB = 16,31 switched on	

Table 3: Internal variables storing polarisation information

Variable		Description	Com. Block
MINT(311)		beam helicity	PYINT1
	=0:	unpolarised	
	=1:	positive helicity	
	=2:	negative helicity	
MINT(312)		target helicity	PYINT1
	=0:	unpolarised	
	=1:	positive helicity	
	=2:	negative helicity	
MINT(313)		helicity of shower initiator on beam side	PYINT1
	=0:	unpolarised	
	=1:	positive helicity	
	=2:	negative helicity	
MINT(314)		helicity of shower initiator on target side	PYINT1
	=0:	unpolarised	
	=1:	positive helicity	
	=2:	negative helicity	
MINT(315)		helicity of hard interacting parton on beam side	PYINT1
	=0:	unpolarised	
	=1:	positive helicity	
	=2:	negative helicity	
MINT(316)		helicity of hard interacting parton on target side	PYINT1
	=0:	unpolarised	
	=1:	positive helicity	
	=2:	negative helicity	
MSTP(179)		switch off polarisation temporarely in PYSIGH and PYSTFU resp.	PYPARS
	=0:	no action	
	=1:	switch off polarisation	

Table 3: Internal variables storing polarisation information, continued

Variable		Description	Com. Block
ISIGH(1000,6)		hard scattering information of Ith line	PYINT3
ISIGH(I,1)		particle code of Ith line on beam side	
ISIGH(I,2)		particle code of Ith line on target side	
ISIGH(I,3)		colour flow	
ISIGH(I,4)		helicity of Ith line on beam side	
ISIGH(I,5)		helicity of Ith line on target side	
ISIGH(I,6)		not used	
KD(I)		polarisation/helicity of Ith line	DPYPOL
	=0:	no polarisation/helicity	
	=1:	positive polarisation/helicity	
	=2:	negative polarisation/helicity	
XSFX(2,-40:40,0:2)		x times parton distribution for given x and Q^2 of flavour KFL = -40:40 and helicity KFLD = 0:2 on beam side (JT=1) and target side (JT=2) resp.	PYINT3
XSFX(JT,KFL,0)		unpolarised	
XSFX(JT,KFL,1)		positive helicity	
XSFX(JT,KFL,2)		negative helicity	

3 Common Blocks and Subroutines

In the following the modified and the new subroutines that have been specifically created for SPHINX are described in more detail. The general tasks of the subroutines themselves as well as the unchanged parameters and variables are not explained, because they are the same as in PYTHIA. The reader is asked again if occasion arises to consult [1] to obtain the needed information. We restrict our explanation to the new aspects in SPHINX. The purpose of the modifications is indicated and the new parameters, switches, and internal variables are listed. Meaning and possible values of the new parameters are given in Table 1. To incorporate polarisation the following common blocks have been enlarged and replace the corresponding PYTHIA common blocks or are added:

- `COMMON/PYINT3/XSFX(2,-40:40,0:2),ISIG(1000,6),SIGH(1000)`
- `COMMON/PYSUBS/MSEL,NSEL,MSUB(200),NSUB(200),KFIN(2,-40:40), CKIN(200)`
- `COMMON/DPYPOL/KD(4000)`

Information about the new internal variables and enlarged arrays can be found in Table 1. In addition it is shown how the local polarisation switch `IPOL` is built up in the different subroutines. Only the polarised case (`IPOL=1`) will be discussed, because in the unpolarised case (`IPOL=0`) each subroutine works exactly as the corresponding PYTHIA subroutine. The not mentioned subroutines of SPHINX are the same as in PYTHIA.

`MAIN PROGRAM`
Purpose: to set up the polarised event generation. The variables which have to been set are listed in Table 1.
Remarks: Examples of a main program are given in Section 4.

`SUBROUTINE PYINIT(FRAME,BEAM,TARGET,WIN)`
Purpose: to display SPHINX header; to check partially the availability of the desired polarisation scenario, i. e. to control that the master switch for polarisation `MSTP(180)` is set properly, the selected partonic subprocesses can be treated polarised and to control and compose the polarisation menue

via `NSEL`; to call `DPLIST` instead of `LULIST` (see below).
New parameters: `MSTP(180)`, `NSEL`, `NSUB(ISUB)`
Internal Polarisation switch: `IPOL=MSTP(180)`
Remarks: If a not allowed scenario has been chosen, the programs stops with an appropriate error message.

`SUBROUTINE PYEVNT`
Purpose: to start polarised event generation; to call `DPEDIT` instead of `LUEDIT` (see below).
New parameters: `MSTP(180)`
Internal polarisation switch: `IPOL=MSTP(180)`

`SUBROUTINE PYINKI(CHFRAM,CHBEAM,CHTARG,WIN)`
Purpose: to check availability of the desired hadronic polarisation scenario, i. e. to control that the selected hadron can be treated polarised and to verify that the polarisation is longitudinal; to store the polarisation of beam and target for the event listing in `KD(1)` and `KD(2)` and for internal use in `MINT(311)` and `MINT(312)`.
New parameters: `KD(I)`, `MSTP(171)`, `MSTP(172)`, `MSTP(180)`
New internal variables: `MINT(311)`, `MINT(312)`
Internal polarisation switch: `IPOL=MSTP(180)`
Remarks: At present only nucleons and hyperons and their antiparticles can be treated polarised. If a not allowed scenario has been chosen, the programs stops with an appropriate error message.

`SUBROUTINE PYRAND`
Purpose: to adapt `PYRAND` to the new environment – all relevant arrays which have been enlarged or added to the common blocks in other subroutines are modified here as well; to extend event shape selection to incorporate helicities; to store helicities of the partons entering the hard interaction according to

- `MINT(313)`: helicity of the beam parton for use in the initial state showering subroutine;

- `MINT(314)`: helicity of the target parton for use in the initial state showering subroutine;

- `MINT(315)`: helicity of the beam parton;
- `MINT(316)`: helicity of the target parton.

New parameters: `MSTP(180)`, `NSUB(ISUB)`
New internal variables: `MINT(313)`, `MINT(314)`, `MINT(315)`, `MINT(316)`
Internal polarisation switch: `IPOL=MSTP(180)`$\times$`NSUB(ISUB)`
Remarks: Note that `MINT(313)=MINT(315)` and `MINT(314)=MINT(316)` but the values of `MINT(313)` and `MINT(314)` are changed later by the initial state shower in `PYSSPA`.

`SUBROUTINE PYSCAT`
Purpose: to adopt `PYSCAT` to the new environment (see `PYRAND`); to store helicities of the partons entering the hard interaction; to fill lines 1, 2 and 5, 6 in the event listing with polarisation information (see below).
New parameters: `KD(I)`, `MSTP(180)`, `NSUB(ISUB)`
New internal variables: `MINT(315)`, `MINT(316)`
Internal polarisation switch: `IPOL=MSTP(180)`$\times$`NSUB(ISUB)`

`SUBROUTINE PYSSPA(IPU1,IPU2)`
Purpose: to perform polarised initial state showering, helicity dependent GLAP evolution equations are used in the backward evolution algorithmus; to enlarge all relevant array in an appropriate manner to incorporate polarisation; to check proper selection of the polarised initial state shower scenario, i. e. to control that the `MSTP(175)` and `MSTP(176)` are set correctly; to store the helicities of initial state shower initiators (`MINT(313)`, `MINT(314)`).
New parameters: `KD(I)`, `MSTP(171)`, `MSTP(172)`, `MSTP(175`, `MSTP(176)`, `MSTP(180)`, `NSUB(ISUB)`
New internal variables: `MINT(313)`, `MINT(314)`, `MINT(315)`, `MINT(316)`
Internal polarisation switch: `IPOL=MSTP(180)`$\times$`NSUB(ISUB)`$\times$`MSTP(176)`
Remarks: At the present stage only QCD shower can be treated polarised, QED showering has to be done in the unpolarised manner. The combination `MSTP(175)=0` and `MSTP(176)=1` allows to simulate *polarised* showering with the use of *unpolarised* parton distributions. This option is just for testing and should not be selected by the user! If `MSTP(175)` or `MSTP(176)` are set

improperly, the programs stops with an appropriate error message. The internal variables MINT(313) and MINT(314) are changed to their final values in this subroutine.

SUBROUTINE PYMULT(MMUL)
Purpose: to switch off polarisation in PYSIGH (set MSTP(179)=1 temporally) when called from PYMULT even in a polarised run, because multiple interaction cannot be treated polarised at the moment.
New parameters: MSTP(179)

SUBROUTINE PYREMN(IPU1,IPU2)
Purpose: to adopt PYREMN to the new environment (see PYRAND); to fill lines 3, 4 in the event listing with polarisation information (see below).
New parameters: KD(I)

SUBROUTINE PYSIGH
Purpose: to evaluate the helicity dependent hadronic cross sections by convolution of the helicity dependent parton distributions with the helicity dependent partonic cross sections; to supply the subroutine with the helicity dependent partonic cross sections.
New parameters: MSTP(171), MSTP(172), MSTP(179), MSTP(180), NSUB(ISUB)
Internal Polarisation switch:
IPOL=MSTP(180)×NSUB(ISUB)×(1-MSTP(179))
Remarks: PYSIGH will always run in the unpolarised mode when it is called by PYMULT which sets MSTP(179)=1 in PYSIGH temporally. Evaluating the spindependent hadronic cross sections one has to notice that the hadrons are specified according to the spin, whereas the partons are labelled by their helicities. The spin is defined relative to the collision axis and the beam is assumed to move in the positive direction. For that reason the helicities at the target side are opposite to the polarisations. Hence the target labels are reversed in the convolution in comparison to the beam labels. The parton distributions are passed from PYSTFU to PYSIGH through the array XPQ(KFL,KFLD) (see below) and stored in the array XSFX(N,KFL,KFLD), where KFLD denotes the helicity. ISIG(N,I) contains the information about the Nth-line in the event listing. The new entries ISIG(N,4) and ISIG(N,5) store the helicities of the

partons at the beam and target side respectively. `ISIG(N,6)` is reserved but not used at the moment.

`SUBROUTINE PYSTFU(KF,X,Q2,XPQ)`

Purpose: to evaluate the helicity dependent parton distributions for given flavour (`KF`), x (`X`), and Q^2 (`Q2`) according to the selected parametrisations.

New parameters: `MSTP(177)`, `MSTP(178)`, `MSTP(179)`, `MSTP(180)`, `NSUB(ISUB)`

Internal Polarisation switch: `IPOL=MSTP(180)`×`(1-MSTP(179))`

Remarks: Call of `PYSTFU` returns x times the parton distribution functions for given flavour, x, and Q^2 for both helicities and an averaged (unpolarised) value. The values are stored in the array `XPQ(KFL,KFLD)` which has been enlarged from `XPQ(-25:25)` to `XPQ(-25:25,0:2)`. Row `XPQ(KFL,0)` contains the unpolarised distributions, `XPQ(KFL,1)` distributions corresponding to the positive helicity (relative to the hadron) and `XPQ(KFL,2)` distributions for the negative helicity. The parton distributions are selected by switches described earlier (see Table 1). The polarised distributions $q_{\pm}$, $g_{\pm}$ are constructed from the unpolarised ones q, g (selected by old PYTHIA switches) and polarised parts Δq and Δg selected by `MSTP(177)` (see Table 1). Four subroutines have been added to calculate the polarised distributions. These are `ALTSTI`[2], `ROSROA`, `ROSROD`, and `GEHSTI`[3]. The subroutines require data files `altsti.dat`, `rosroa.dat`, and `partons.dat`. These files must be visible to `FORTRAN` `open` statement and therefore they have to be placed in appropriate directory. The files are supplied with the program. The parton distribution for fixed helicity are reconstructed from the polarised and unpolarised distributions according to $q_{\pm} = \frac{1}{2}(q \pm \Delta q)$. When the polarised and unpolarised parts are combined together the program performs the unitarity check – if the resulting total distribution becomes negative for one helicity it is put to zero and the corresponding result for the other helicity is set to the value of the unpolarised part. Only polarised parametrisations for protons are implemented. Neutron parametrisations are obtained from them by isospin symmetry, the parametrisations for hyperons are constructed by naive SU(3).

[2]This subroutine has been written by G. Altarelli and J. Stirling. Used with permision from the authors.

[3] The three parametrisation of Gehrmann&Stirling contained in `GEHSTI` are brandnew and not thouroughly tested yet.

Charge conjugations is used to describe the corresponding antiparticles.

`SUBROUTINE ALTSTI(X,Q2,UPV,DNV,SEA,STR,CHM,BOT,TOP,GLU)`

Purpose: to return x times the polarised parton distributions evaluated at given x (`X`) and Q^2 (`Q2`) according to the parametrisation of Altarelli&Stirling [3]. `UPV` denotes the valence distribution of up-quarks $x\Delta u^{\mathrm{val}}(x,Q^2)$, `DNV` for down-quarks $x\Delta d^{\mathrm{val}}(x,Q^2)$. `SEA` signifies the sea distribution $x\Delta q^{\mathrm{sea}}(x,Q^2)$. `STR`, `CHM`, `BOT`, and `TOP` label the distributions for the strange-, charm-, bottom-, and top-quark $x\Delta q(x,Q^2)$, $q = s,c,b,t$ respectively. Finally `GLU` marks the gluon distribution $x\Delta g(x,Q^2)$.

Remarks: `ALTSTI` requires the data file `altsti.dat` which has to be placed in an appropriate directory. `altsti.dat` is supplied with this program.

`SUBROUTINE ROSROD(X,Q2,XPDF)`

Purpose: to return x times the polarised parton distributions evaluated at given x (`X`) and Q^2 (`Q2`) according to the parametrisation of Ross&Roberts set d [4]. `XPDF(-6:6)` contains $x\Delta q(x,Q^2)$ for $q = \bar{t},\bar{b},\bar{c},\bar{s},\bar{u},\bar{d},g,d,u,s,c,b,t$ in this order.

`SUBROUTINE ROSROA(X,Q2,XPDF)`

Purpose: to return x times the polarised parton distributions evaluated at given x (`X`) and Q^2 (`Q2`) according to the parametrisation of Ross&Roberts set a [4]. The contents of `XPDF(-6:6)` is explained above.

Remarks: `ROSROA` requires the data file `rosroa.dat` which has to be placed in an appropriate directory. `rosroa.dat` is supplied with this program.

`SUBROUTINE GEHSTI(IGFLAG,X,Q2,XDDPR)`

Purpose: to return x times the polarised parton distributions evaluated at given x (`X`) and Q^2 (`Q2`) according to the parametrisation of Gehrmann&Stirling set a (`IGFLAG=0`), set b (`IGFLAG=1`) or set c (`IGFLAG=2`). [6]. `XDDPR(-6:6)` contains $x\Delta q(x,Q^2)$ for $q = \bar{t},\bar{b},\bar{c},\bar{s},\bar{u},\bar{d},g,d,u,s,c,b,t$ in this order.

Remarks: `GEHSTI` requires the program package `parton.f`, consisting of the subroutines `polpar`, `parini`, `getpar`, `q2low`, `q2high`, `xlow`, and `xhigh` and the data file `partons.dat` written by Gehrmann and Stirling and used here with permission of the authors. For further informations about this package see [6]. `partons.dat` has to be placed in an appropriate directory

and is supplied with this program. The package `parton.f` is contained in this program.

`SUBROUTINE DPLIST(MLIST)`
Purpose: to display the polarisations of the particle in the event listing.
New parameters: `KD(I)`
Remarks: `DPLIST` is a modification of the JETSET subroutine `LULIST`. It is changed to display the polarisation in the final listing. The sign displayed just behind the particle code denotes polarisation with respect to the z-axis. When the sign is missing the particle has been treated as unpolarised. The information is taken from the vector `KD(I)` and transformed accordingly to (‘0’,‘1’,‘2’)→(‘ ’,‘+’,‘−’). The following format is chosen (polarisation for the colliding hadrons, helicity for the partons resp.):

- = ‘ ’: no polarisation/helicity
- = ‘+’: positive polarisation/helicity
- = ‘−’: negative polarization/helicity

`SUBROUTINE DPEDIT(MEDIT)`
Purpose: to compress the vector `KD(I)`, containing the polarisation information, properly.
New parameters: `KD(I)`
Remarks: `DPEDIT` is a modification of the JETSET subroutine `LUEDIT`.

4 Examples

In the following we give two examples of a main program for a simulation with SPHINX and show the corresponding results. We considered longitudinal polarised proton-proton scattering in the CMS at $\sqrt{s}$ = 200 GeV and selected the process $qg \to qg$ for the partonic interaction. In the first example both beam and target are polarised in $+z$-direction, in the second example the target spin is reversed. With regard to the event listings is has to be mentioned that the displayed format differs from the real because we

removed a few columns such that it fits in this text. In addition the event listing is cut after the hard interaction (denoted by $\cdots$), i. e. behind line 8. In the omitted part there is no polarisation information and it has the same format as the original PYTHIA listing.

The information about the polarisation flow is displayed as explained above right behind the flavour code `KF`. The first row contains the information about the beam particle. In addition to the PYTHIA labels the sign '+' behind the flavour code for the proton `KF=2212` denotes the polarisation in positive z-direction. Accordingly the sign '+' ('−') in the second row signifies the polarisation of the target in positive (negative) z-direction in the first (second) example. The third and fourth line represents the initiators of the initial state shower. In both examples they are gluons with positive helicities for both the beam and the target side. During the initial state shower the beam side parton becomes an $\bar{s}$ with positive helicity, whereas the target side parton remains a positive helicity gluon. These partons undergo the hard interaction the resulting partons of which are displayed in the lines seven and eight. In the following the final state interaction takes place, i. e. the outgoing partons fragment, the unstable produced hadrons decay, etc. as long as only stable particles exist. In the final state polarisation is not traced and consequently there is no polarisation informations about these lines provided. This part of the listing is then again the same as the corresponding PYTHIA listing.

4.1 The Main Programs

4.1.1 First Example – parallel polarisation

```
C___    Example of a Main Program for event generating
C___    in longitudinal polarised proton-proton-scattering
C___
C___    E.g.: polarised p(+)p(+)-scattering in CMS
C___          at sqrt(s)=200 GeV
C___
C___    This program has to be linked with
C___    the programs SPHINX and JETSET7.4,
C___    the data files ALTSTI.DAT and ROSROA.DAT,
C___    and the CERN-Libraries.
```

```
C___    COMMON BLOCKS of SPHINX for event generation

        COMMON/LUDAT1/MSTU(200),PARU(200),MSTJ(200),PARJ(200)
        COMMON/LUJETS/N,K(4000,5),P(4000,5),V(4000,5)
        COMMON/PYSUBS/MSEL,NSEL,MSUB(200),NSUB(200),
     &  KFIN(2,-40:40),CKIN(200)
        COMMON/PYPARS/MSTP(200),PARP(200),MSTI(200),PARI(200)
        COMMON/PYINT5/NGEN(0:200,3),XSEC(0:200,3)

C===============================================================
C       Polarisation Set-Up
C===============================================================

C___    Polarised simulation
        MSTP(180)=1

C___    Beam positive polarised
        MSTP(171)=1

C___    Target positive polarised
        MSTP(172)=1

C___    Polarised parton distributions a la Altarelli&Stirling
        MSTP(177)=2

C___    Polarised Initial State Shower
        MSTP(176)=1
C___    with polarised parton distributions
        MSTP(175)=1

C===============================================================

C===============================================================
C       Event Set-Up
C===============================================================
```

```
C___    Choice of parton processes ``a la carte''
        MSEL=0

C___    Choice of polarised parton processes ``a la carte''
        NSEL=0

C___    Choice of process: qg --> qg
        MSUB(28)=1

C___    Process 28 polarised
        NSUB(28)=1

C___    Number of generated events
        NEVENT=1000

C___    kinematical cuts
C___    P_T-cut (minimum)
        CKIN(3)=3.
C___    P_T-cut (maximum)
        CKIN(4)=25.

C================================================================

C================================================================
C       Start of event generation
C================================================================

C___    Initialisation

        CALL PYINIT('CMS','p','p',200.)

C___    Loop over events
        DO 100 I=1,NEVENT

C___    Event generation
          CALL PYEVNT
```

```
C___    List the first event with spininformation
          IF(I.EQ.1) CALL DPLIST(1)

  100     CONTINUE

C___    Print cross section and histogram
        CALL PYSTAT(1)

        END
```

4.1.2 Second Example – antiparallel polarisation

The second example is constructed by replacing `MSTP(172)=1` by `MSTP(172)=2` in the first example, i. e. by switching the spin of the target.

4.2 The Event Listings

4.2.1 First example – parallel polarisation

```
                              SPHINX
            **  Last date of change:   6 Aug 1994  **

            The Lund Monte Carlo - PYTHIA version 5.7
            **  Last date of change:   3 Apr 1992  **

            The Lund Monte Carlo - JETSET version 7.4
            **  Last date of change:  10 Mar 1992  **

1********** PYINIT: initialization of PYTHIA routines **********

 ==============================================================
 I                                                            I
 I      PYTHIA will be initialized for a p on p collider      I
 I          at    200.000 GeV center-of-mass energy           I
 I                                                            I
 ==============================================================
```

```
*PYMAXI: summary of differential cross-section maximum search *

  ==========================================================
  I                                      I                 I
  I  ISUB  Subprocess name               I  Maximum value  I
  I                                      I                 I
  ==========================================================
  I                                      I                 I
  I   28   f + g -> f + g                I    3.9475E+00   I
  I   96   Semihard QCD 2 -> 2           I    1.7380E+02   I
  I                                      I                 I
  ==========================================================

************** PYINIT: initialization completed ***************


                  Event listing (summary)

   I  particle/jet KF orig   p_x     p_y     p_z      E      m

   1  !p+!       2212+   0   0.000   0.000  99.996 100.000  0.938
   2  !p+!       2212+   0   0.000   0.000 -99.996 100.000  0.938
=================================================================
   3  !g!          21+   1   0.130  -0.241  33.098  33.099  0.000
   4  !g!          21+   2   0.151  -0.503  -3.866   3.902  0.000
   5  !s~!         -3+   3  -0.838  -0.101  27.030  27.043  0.000
   6  !g!          21+   4  -0.036   2.481  -0.350   2.505  0.000
   7  !s~!         -3    0   2.361   2.098  26.049  26.240  0.199
   8  !g!          21    0  -3.235   0.281   0.631   3.308  0.000
=================================================================
   ...

1** PYSTAT:  Statistics on Number of Events and Cross-sections ***

=================================================================
I                              I                    I          I
```

```
I       Subprocess              I  Number of points  I   Sigma   I
I                               I                    I           I
I-------------------------------I--------------------I    (mb)   I
I                               I                    I           I
I N:o Type                      I Generated    Tried I           I
I                               I                    I           I
=================================================================
I                               I                    I           I
I  0 All included subprocesses I      1000     5855 I 6.695E-01 I
I 28 f + g -> f + g             I      1000     5855 I 6.695E-01 I
I                               I                    I           I
=================================================================

** Fraction of events that fail fragmentation cuts =  0.00000 ***
```

4.2.2 Second example – antiparallel polarisation

```
                          SPHINX
            **  Last date of change:   6 Aug 1994  **

            The Lund Monte Carlo - PYTHIA version 5.7
            **  Last date of change:   3 Apr 1992  **

            The Lund Monte Carlo - JETSET version 7.4
            **  Last date of change:  10 Mar 1992  **

1********** PYINIT: initialization of PYTHIA routines **********

 ===============================================================
 I                                                             I
 I      PYTHIA will be initialized for a p on p collider       I
 I          at    200.000 GeV center-of-mass energy            I
 I                                                             I
 ===============================================================

 *PYMAXI: summary of differential cross-section maximum search *
```

```
 ==============================================================
 I                                       I                    I
 I  ISUB  Subprocess name                I  Maximum value     I
 I                                       I                    I
 ==============================================================
 I                                       I                    I
 I   28   f + g -> f + g                 I    3.9772E+00      I
 I   96   Semihard QCD 2 -> 2            I    1.7380E+02      I
 I                                       I                    I
 ==============================================================

 ************** PYINIT: initialization completed ***************
```

```
                    Event listing (summary)

    I  particle/jet KF orig   p_x     p_y     p_z      E      m

    1  !p+!       2212+   0   0.000   0.000  99.996 100.000  0.938
    2  !p+!       2212-   0   0.000   0.000 -99.996 100.000  0.938
 ==================================================================
    3  !g!          21+   1   0.784   0.462  33.089  33.102  0.000
    4  !g!          21+   2  -0.502   0.119 -33.887  33.891  0.000
    5  !s~!         -3+   3  -0.240   0.737  21.851  21.865  0.000
    6  !g!          21+   4  -5.178  -4.192   4.698   8.152  0.000
    7  !s~!         -3    0  -4.508   0.563  25.310  25.715  0.199
    8  !g!          21    0  -0.910  -4.018   1.239   4.302  0.000
 ==================================================================
    ...

1** PYSTAT:  Statistics on Number of Events and Cross-sections ***

 ==================================================================
 I                              I                    I            I
 I        Subprocess            I  Number of points  I   Sigma    I
 I                              I                    I            I
```

```
I------------------------------I--------------------I    (mb)   I
I                              I                    I           I
I N:o Type                     I Generated    Tried I           I
I                              I                    I           I
=================================================================
I                              I                    I           I
I  0 All included subprocesses I      1000     5870 I 6.868E-01 I
I 28 f + g -> f + g            I      1000     5870 I 6.868E-01 I
I                              I                    I           I
=================================================================

** Fraction of events that fail fragmentation cuts =  0.00000 ***
```

Acknowledgements

We all thank T. Sjöstrand very much for his help and encouragement in understanding the details of the PYTHIA code. A. S. thanks the MPI für Kernphysik for its support. The work of L. M. was supported in part by KBN under grant 2 P302 143 06. This work was supported also by DFG (Scha 458/3-1) and BMFT-KBN (Projekt X081.92)

References

[1] T. Sjostrand, Computer Phys. Commun. 82 (1994) 74

[2] *Proc. Seventh Int. Conf. in Ultra-Relativistic Nucleus-Nucleus Collisions*, Eds. G. Baym *et al.*, Nucl. Phys. **A498** (1989); RHIC Spin Collaboration, *Proposal on Spin Physics Using the RHIC Polarized Collider* August 1992;
RHIC Spin Collaboration, UPDATE, Oktober 1993

[3] G. Altarelli and J. Stirling, Particle World 1 (1989) 40.

[4] G. G. Ross and R. G. Roberts, Rutherford preprint RAL-90-062 (1990).

[5] G. Altarelli and G. Parisi, Nucl. Phys. **B175** (1977) 298

[6] T. Gehrmann and W. J. Stirling, Durham preprint DTP/94/38 (1994).

[7] R. Gastmans and Tai Tsun Wu, *The Ubiquitous Photon – Helicity Method for QED and QCD*, Oxford Science Publications (1990); K. Hidaka Nucl. Phys. **B192** (1981) 369

[8] H.-U. Bengtsson and T. Sjöstrand, Computer Phys. Commun. 46 (1987) 43.